Departamento de Informática - Faculdade de Ciências da Universidade de Lisboa

CELPASIM 2.0
Manual do Utilizador

Versão 2.0.0.(draft)

André Falcão

Agosto de 2009

Índice

CELPASIM 2.0 MANUAL DO UTILIZADOR....1

ÍNDICE....2

1. INTRODUÇÃO....4

2. ENTRADA DE DADOS....5

2.1. REQUISITOS DO PROGRAMA....5
2.2. ENTRADA DE DADOS....5

3. PARÂMETROS DE SIMULAÇÃO....7

3.1. PARÂMETROS GENÉRICOS DE SIMULAÇÃO....7
3.2. PROBABILIDADES DE INCÊNDIOS....8
3.3. ORDENAÇÃO DE POVOAMENTOS....9
3.4. NÍVEIS ANUAIS DE PROCURA....10
3.5. ÁREAS NOVAS....11

4. SIMULAÇÃO....13

4.1. PROCEDIMENTO DE SIMULAÇÃO....13
4.2. INICIALIZAÇÃO DE VARIÁVEIS....14
4.3. SIMULAÇÃO DO CRESCIMENTO....14
4.3.1. Estimação da área basal....15
4.3.2.Estimação da altura dominante....15
4.3.3 Estimação do volume....15

5. APRESENTAÇÃO DE RESULTADOS....16

5.1.RESULTADOS NO ECRÃ....16
5.2. RESULTADOS EM FICHEIRO....17
5.2.1. Informação geral - geral.csv....17
5.2.2. Simulação ao nível do concelho - concelhos.csv....18
5.2.3. Simulação ao nível da área de gestão - sim_detalhe.csv....19
5.2.4. Áreas ardidas - fogos.csv....19
5.2.5. Áreas cortadas - areas_cort.csv....20
5.2.6. Áreas novas - areas_novas.csv....20
Agradecimentos:....20

APÊNDICE - CÓDIGO FONTE DO PROGRAMA CELPASIM 1.0.0....21

A.1. MÓDULO GLOBAIS.BAS....21
A.2. MÓDULO OTHER.BAS....28

A.3. MÓDULO DE ENTRADA E SAÍDA.....35

1. Introdução

O programa CELPASIM foi um esforço conjunto da CELPA e do Instituto Superior de Agronomia para produzir uma ferramenta que possibilitasse a simulação das áreas de eucaliptal nacionais que não estão sob o controlo directo das empresas.

O programa funciona como um simulador de crescimento, definido as regras de corte por um sistema de ordenação, não incluindo portanto qualquer tentativa de optimização.

O CELPASIM oferece um conjunto vasto de opções de controle que permitem criar e manipular diferentes situações de gestão. Nomeadamente:

- Possibilidade de definir diferentes cenários de procura de material lenhoso
- Inclusão de probabilidades de incêndio diferenciadas por regiões
- Definição e alteração dos critérios de ordenação dos povoamentos para corte
- Introdução de áreas novas por ano e região

A ferramenta desenvolvida pode ser executada em qualquer PC de uso corrente sendo muito rápida a realização de qualquer simulação. É possível testar diferentes cenários em minutos, correndo cada simulação em menos de um minuto com recursos computacionais modestos. Os resultados produzidos podem ser lidos directamente a partir de qualquer folha de cálculo, como o Microsoft Excel, permitindo avançadas capacidades de análise.

Este manual cobre a versão 1.0.0 do simulador

2. Entrada de dados

2.1. Requisitos do programa

O CELPASIM foi desenvolvido em Visual Basic 6.0 e pode ser executado em qualquer máquina a correr o sistema operativo Windows em 32 bits, nomeadamente:

- Windows 95/98/ME
- Windows 2000
- Windows XP Home Edition/Pro

Como requisitos mínimos, o computador deverá ter um processador compatível com Intel Pentium ou superior a correr a 200 MHz e 64 MBytes de RAM e pelo menos 30 MB de espaço livre em disco. É desejável contudo que a velocidade do processador ultrapasse 1.0 GHz, para minimizar o tempo de simulação.

2.2. Entrada de dados

Os dados devem ser fornecidos ao programa sob a forma de um ficheiro com a extensão CSV (*comma separated values*). Estes ficheiros podem ser produzidos a partir de qualquer folha de cálculo, nomeadamente o Microsoft Excel. A utilização deste tipo de ficheiros obriga a algumas configurações específicas da máquina de trabalho(Painel de Controle >> Opções Regionais) da seguinte forma:

- Separador de dígitos decimais - "." (ponto)
- Separador de colunas - "," (vírgula)

- Separador de 'milhares' - " " (espaço)

A tabela CSV pode ter o nome que se deseje, mas o programa procura por omissão um ficheiro com a designação '*work.csv*'. A tabela deve ter a seguinte configuração **rígida**. Uma primeira linha com os títulos das variáveis seguida de tantas linhas quantos os povoamentos a considerar, cada uma com 13 colunas (Tabela 2.1)

Tabela 2.1 - estrutura da tabela com os dados de entrada

Nº	Designação	Tipo	Descrição
1	Parcela	text	Identificador alfanumérico da parcela
2	conc_code	int	Código do concelho
3	cod_reg	int	Código da região CELPA
4	estrato_idade	int	Idade aproximada do estrato
5	area	float	área correspondente da parcela
6	rot	int	Rotação
7	n_peqs	int	Nº de árvores com DAP < 5 cm
8	n_grnds	int	Nº de árvores com DAP >= 5 cm
9	hd	float	Altura dominante
10	ab	float	Área basal por ha
11	dqm	float	Diâmetro quadrático médio
12	IQE	float	Indíce de qualidade da estação
13	v5sch	float	Volume das árvores com dap >= 5 cm

3. Parâmetros de simulação

O CELPASIM dispõe de 7 interfaces distintos para possibilitar diferentes tipos de simulação. É possível alterar virtualmente quase todos os parâmetros que condicionam o processo de simulação. De uma forma genérica podem considerar-se 5 grandes classes de parâmetros descritas a seguir:

1. Parâmetros genéricos de simulação
2. Probabilidades de incêndio
3. Ordenação de povoamentos para corte
4. Níveis anuais de procura de material lenhoso
5. Áreas novas a plantar

3.1. Parâmetros genéricos de simulação

O interface com os parâmetros genéricos de simulação apresenta duas áreas principais. A primeira é referente ao horizonte temporal da simulação, onde o utilizador pode definir qual o período de simulação e qual o ano de referência para o qual vai executar a simulação. Seguidamente aparecem vários parâmetros que permitem condicionar diferentes aspectos da simulação e da gestão:

Probabilidade de abandono: Este parâmetro indica a proporção da área que, após o corte em 3ª rotação (passagem para a 4ª), deverá ser abandonada

Probabilidade de reconversão: Indica qual a percentagem de áreas cortadas que são reconvertidas para 1ª rotação.

Número de varas por toiça: Indica o número médio de varas a deixar por toiça após o corte final.

Este interface permite ainda aceder às probabilidades de incêndio por região através de um botão.

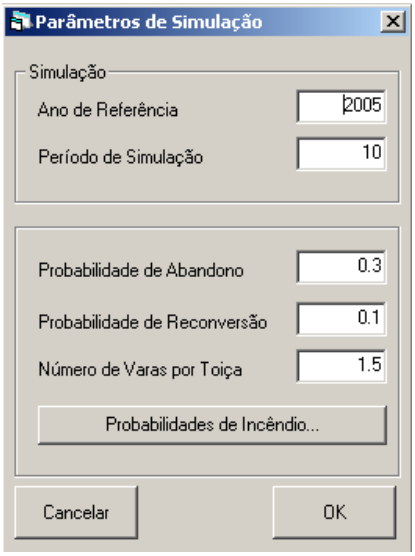


Figura 3.1 Parâmetros genéricos de simulação

O abandono e a reconversão são probabilidades fixas aplicadas a todas as áreas que entram em corte final. Primeiro são processados os abandonos e, á área que fica, é sujeita a reconversão. Com os valores expostos, se uma parcela com 100 ha em 3ª rotação é cortada, 30 ha são abandonados. Dos 70 ha que não são abandonados, 7 ha reconvertidos para 1a rotação.

Áreas divididas pelos cortes, por se ter atingido o volume de procura pretendido, são identificadas na simulação pelo código "CT". Assim estas parcelas são marcadas como:

CT_PPPP_AA

em que PPPP representa o código da parcela que originou o corte e AA o ano em que este for cortado.

Áreas reconvertidas também originam áreas novas, utilizando o prefixo "REC". Deste modo:

REC_PPPP_AA

em que, à semelhança do caso anterior, PPPP representa o código da parcela antes da reconversão e AA o ano em que ocorreu

3.2. Probabilidades de incêndios

A probabilidade de incêndio para cada região determina qual a quantidade de área de eucaliptal que será simulada como ardida anualmente, ao longo do período de

simulação. É inicializada com um valor de 5 % indicando que sairá 5 % da área total de todos os povoamentos independentemente da idade

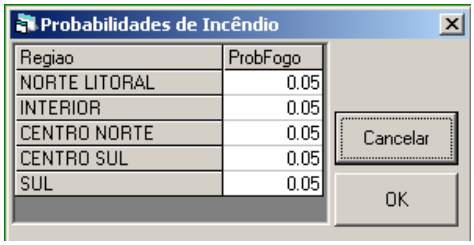


Figura 3.2. Interface de probabilidade de incêndios

O volume ardido é contabilizado separadamente e apresentado numa tabela à parte (ver 5.2.4). As áreas ardidas são acumuladas ao nível do concelho. Caso esse valor por concelho seja inferior a 10 ha, então essa área é abandonada da simulação. Áreas ardidas são consideradas como estando em 2ª rotação com idade 0. São assim originadas várias parcelas novas na decorrência de fogo. Na simulação essas parcelas vão ter um identificador novo com o prefixo "FG":

FG_CCC_AA

em que CCC corresponde ao concelho onde estavam instaladas as parcelas que arderam e AA é o ano em que sucedeu o incêndio

3.3. Ordenação de povoamentos

Os povoamentos ou áreas de gestão no processo de simulação são ordenados de acordo com uma avaliação ponderada. São considerados 3 critérios principais para a ordenação de um povoamento:

- Idade
- Região
- Diâmetro quadrático médio

O processo de ordenação decorre somando os pesos atribuídos para cada povoamento e seleccionando para corte aqueles que têm valores mais elevados. Como exemplo, um povoamento situado em Odemira (Sul) com uma idade de 11 anos e um diâmetro quadrático médio (DQM) de 13.1 cm teria um factor de: 3 + 6 + 3 =12.

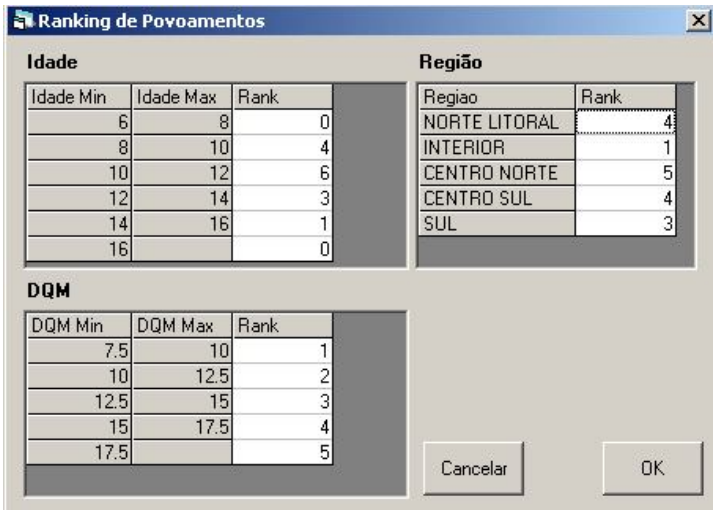


Figura 3.3. Critérios de ordenação dos povoamentos

O interface (Figura 3.3) mostra 3 quadros para cada uma das variáveis, de notar que cada quadro tem um número de colunas fixas, que não podem ser alteradas e que pode ser lido da seguinte forma: O caso da região é o mais simples. A primeira coluna indica a região e a segunda o peso variável atribuído a cada uma. Para a idade a primeira coluna corresponde ao valor mínimo e a segunda à idade máxima (inclusive). Um povoamento com 11 anos está assim na 3ª linha, apresentando um factor de 6. O DQM é usado de acordo com o mesmo princípio. A 3a coluna representa o peso atribuído a cada intervalo de possível, sendo a última linha o peso atribuído aos factores que tenham mais do que o limite definido

3.4. Níveis anuais de procura

A simulação do eucaliptal pressupõe que haja procura da indústria para absorver a madeira cortada. No inteface de *Níveis de Procura* é dado ao utilizador uma forma simples de poder manipular esses valores.

Os volumes são mostrados em 10^3 x m^3. Podem ser alterados de uma forma maunual ou semi automática. Caso se queira atribuir o mesmo valor de procura para todos os anos basta preencher o valor desejado na caixa de texto *valor comum* e pressionar o botão *Aplicar*. A tabela tomará, para todos os anos, o valor escolhido.

A manipulação manual exige um processamento mais elaborado mas bastante mais flexível. Para editar qualquer célula na 2ª coluna da tabela, é suficiente escrever o valor numérico desejado, com as seguintes regras de edição:

1. Só funcionam os números e o ponto. Não é possível introduzir letras.
2. No início da escrita, adiciona valores ao texto presente na célula. Por exemplo se estiver o valor '1000', se introduzir o valor '500', o que ficará registado será '1000500'

3. Para apagar um caracter, mesmo dos que estão previamente em cada célula, basta premir *Backspace*.
4. Para apagar o valor total de uma célula premir *Delete*.

Estes procedimentos de edição são válidos para todas as tabelas presentes nos restantes interfaces do programa.

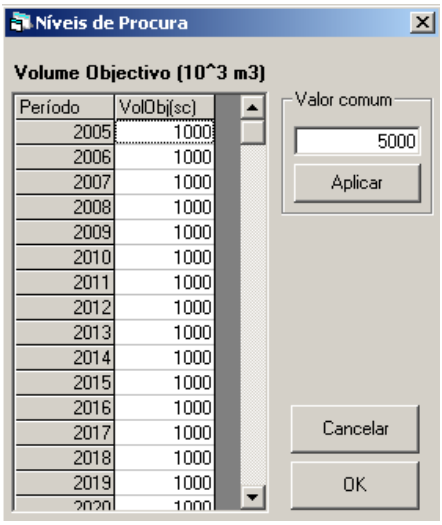


Figura 3.4. Interface de definição de níveis de procura

3.5. Áreas novas

As áreas novas são definidas pelo utilizador antes de se iniciar a simulação, podendo ser escolhida a área para cada região e ano. À semelhança do caso anterior, é possível manipular os valores das células de uma forma semi-automática, bastando para isso escrever um valor comum na caixa correspondente e pressionar o botão *Aplicar*. Os mecanismos de edição manual da tabela são os mesmo descritos no ponto anterior.

As áreas novas são atribuídas de acordo com a proporção de área com eucalipto correspondente a cada concelho para a respectiva região. Por exemplo, supondo que a região R tem 3 concelhos apenas: X, Y e Z, todos com a mesma área. A quantidade de eucalipto presente em cada um deles é que é diferente, correspondendo, no concelho X a 20 % do total de eucalipto presente na Região R, para Y 50 %, tendo Z cerca de 30 %. Supondo uma florestação anual de 500 ha, o concelho X receberá mais 100 ha, Y 250 ha e Z 150 ha.

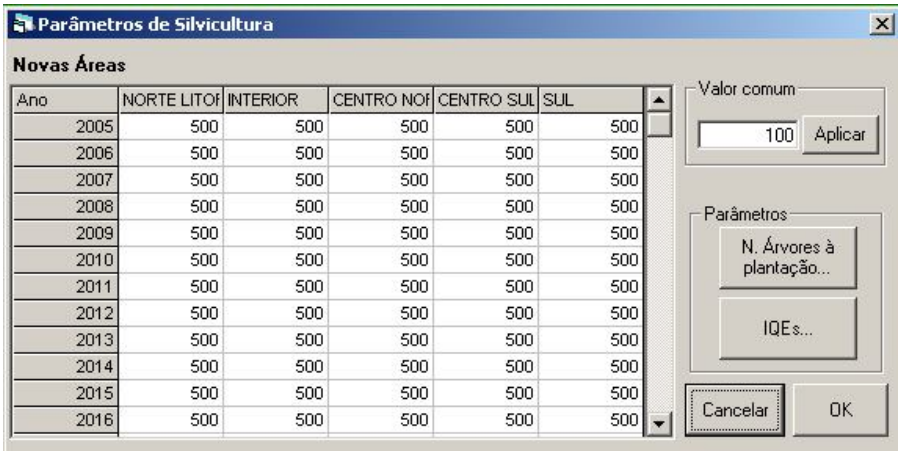


Figura 3.5. Introdução de áreas novas

Neste interface é possível ainda ter acesso a dois outros formulários, um relativo à inicialização do número de árvores à plantação para as áreas novas (Figura 3.6.) e outro relativo aos índices de qualidade da estação (Figura 3.7.).

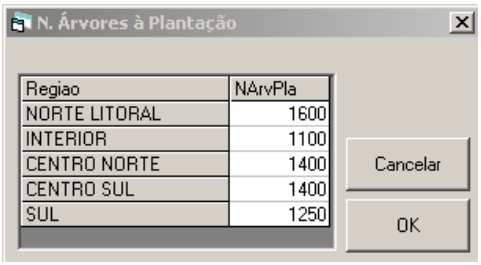


Figura 3.6. Definição do Nº de árvores à plantação para áreas novas

Ambos os formulários permitem definir quais os valores do número de árvores à plantação e o valor por omissão do IQE, para cada área nova

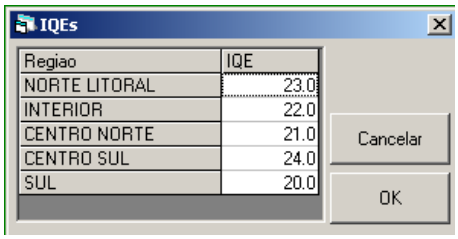


Figura 3.7. Definição do IQE para áreas novas

No procedimento de simulação, a cada área nova em cada concelho é atribuído um código identificador com a seguinte designação:

NA_AA_CCC

NA é o acrónimo de Novas Áreas, AA representa dois dígitos referentes ao período de simulação, CCC é referente ao concelho em que a nova parcela foi instalada.

4. Simulação

4.1. Procedimento de simulação

Para executar uma simulação após a introdução de vários parâmetros, basta, no menu inicial da aplicação seleccionar a opção Simular > Correr Simulação. Aparece então uma caixa de diálogo (Figura 4.1.) que permite ao utilizador criar um directório novo que armazenará toda a informação relevante respeitante à simulação realizada, nomeadamente todos os ficheiros de saída (ver ponto 5.2).

A simulação decorre pela seguinte ordem de operações:

1. Faz crescer a floresta e incrementa o ano
2. Determina as áreas que vão arder e processa-as como tal (ver ponto 3.2)
3. Calcula o stock
4. Executa os cortes e processa os abandonos
5. Processa quais as áreas que vão ser reconvertidas
6. Acrescenta as áreas novas

NOTA: O procedimento de simulação pode ser verificado no código fonte disponível em apêndice. A rotina que coordena todo o processo designa-se de **RunSimulation**, no módulo **other.bas**

O número de parcelas no processo de simulação vai crescendo ao longo do tempo devido a vários factores:

a) Cortes. O programa vai procurar atingir exactamente os volumes pretendidos e definidos nos níveis de procura. Se por acaso o corte da área total de uma parcela

fizer com que o volume já cortado exceda o desejado então essa parcela será dividida. Uma parte continuará a crescer e a outra será cortada, passando para a rotação seguinte

b) Reconversões. No caso da probabilidade de reconversões ser diferente de 0 ou 1, todas as áreas a corte são potencialmente reconvertidas de acordo com o procedimento descrito no ponto 3.1.

c) Fogos. Em cada ano todas as parcelas sofrem uma redução de área pelo fogo (ponto 3.2.). Essa área é acumulada ao nível do concelho a que pertencem, sendo potencialmente criada uma área nova por concelho florestado por ano

d) Áreas novas. São criadas áreas novas em cada concelho em cada período de simulação

4.2. Inicialização de variáveis

As variáveis biométricas mais importantes dos povoamentos e fundamentais para a simulação do cresicomento (área basal, altura dominante e número de árvores por ha devem estar assinaladas para todas as parcelas. Quando isso não sucede é necessário encetar uma série de procedimentos para proceder à sua inicialização com os modelos disponíveis.

Para a área basal e altura dominante, o procedimento é similar. Uma vez que o modelo GYMMA (Barreiro et al. 2004) não permite fazer inicializações de raíz, a solução encontrada foi assumir que a idade estimada para cada povoamento era correcta e utilizar o modelo GLOBULUS (Tomé 2001) para uma primeira estimativa. Durante a simulação, para estas parcelas é utilizado o modelo GYMMA como se a idade não fosse conhecida.

No caso de não haver árvores em povoamentos com idades maiores que 0, o número de árvores do povoamento é inicializado de acordo com a única função disponível. Contudo essa função inicializa o número de árvores total e não o número de árvores maiores que 5, que é o que é necessário para a área basal (modelo GYMMA). Com esta estratégia pode suceder que no caso de parcelas sem árvores, o modelo dará uma estimativa por excesso

4.3. Simulação do crescimento

A simulação do crescimento no programa CELPASIM é feita com base em dois modelos de crescimento. Para parcelas de inventário sem idade precisa conhecida usou-se o modelo GYMMA (Barreiro et al. 2004). Após cortes, fogos ou reconversões, quando a idade passa a ser conhecida, o modelo utilizado é o Globulus 2.0.1 (Tomé 2001).

São ambos modelos de povoamento que se baseiam na estimação da área basal e da altura dominante para o cálculo posterior do volume total.

NOTA: As rotinas do código fonte relativas à simulação do crescimento das diferentes parcelas são **Grow** e **simulate** e estão no módulo **globais.bas**, no apêndice

Não se pretende com este manual descrever os modelos de crescimento utilizados, contudo é importante fazer referência a procedimentos de excepção verificados devido a alguns problemas de estabilidade numérica verificados para algumas situações de inventário.

4.3.1. Estimação da área basal

Na estimação da área basal verificou-se um problema importante. Para certas situações biométricas, os resultados simulados afastavam-se demasiado da realidade observada verificando-se acréscimos demasiado elevados ou valores excessivos de área basal estimada. Para resolver esta situação, ajustou-se um modelo de crescimento novo usando o Software STATISTICA 5.0., usando a idade aproximada das parcelas. O modelo utilizado foi o de McDill e Amateis (1992).

$$ab_2 = \frac{M}{\left(1-\left(1-\frac{M}{ab_1}\right)\left(\frac{t}{t+1}\right)^a\right)}$$

com

M = 63.81254

a = 1.381175

O procedimento de balizagem é o seguinte. Caso a área basal calculada cresça mais de 3 m2 de um ano para o outro ou caso o valor estimado seja superior a a 95 % do valor da assimptota do modelo novo (63.81254) então a área basal é calculada pela nova equação.

4.3.2.Estimação da altura dominante

Verificou-se o aparecimento de erros numéricos também no modelo GYMMA para a altura dominante. Estes sucedem em certas situações em que área basal calculada anteriormente é grande demais e a altura dominante é pequena. Caso isso suceda, um dos parâmetros que entra no modelo da altura dominante fica negativo criando impossibilidades numéricas.

Para lidar com esta situação, optou-se por calcular de antemão o parâmetro em causa. Caso este seja negativo, é executado o modelo globulus utilizando a idade estimada como idade verdadeira .

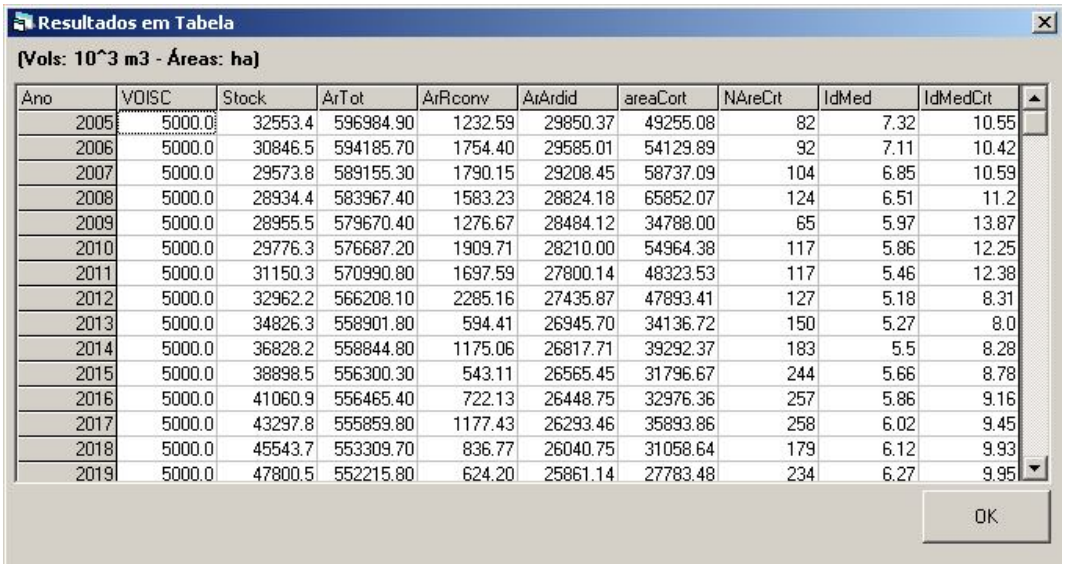
4.3.3 Estimação do volume

O modelo GYMMA possibilita a estimação de volumes com e sem casca, com e sem toiça e permite ainda fazer variar a altura da "bicada". Contudo o modelo GLOBULUS não é tão flexível possibilitando apenas o cálculo do material lenhoso com e sem casca. Para tornar essas duas metodologias comparáveis optou-se por utilizar apenas o modelo GYMMA de cálculo do volume sem casca e sem toiça. Para o GLOBULUS utilizou-se, naturalmente, o modelo que não inclui a casca.

5. Apresentação de resultados

5.1.Resultados no Ecrã

Os resultados de simulação do CELPASIM são exportados automaticamente para ficheiros CSV, lidos directamente por qualquer folha de cálculo como o Microsoft Excel. É contudo dado ao utilizador a possibilidade de analisar um resultado executivo com os principais resultados da simulação numa tabela apresentada depois do o programa executar uma simulação (Figura 5.1.). Os volumes aparecem em milhares de m^3 e as áreas em ha).



Ano	VOlSC	Stock	ArTot	ArRconv	ArArdid	areaCort	NAreCrt	IdMed	IdMedCrt
2005	5000.0	32553.4	596984.90	1232.59	29850.37	49255.08	82	7.32	10.55
2006	5000.0	30846.5	594185.70	1754.40	29585.01	54129.89	92	7.11	10.42
2007	5000.0	29573.8	589155.30	1790.15	29208.45	58737.09	104	6.85	10.59
2008	5000.0	28934.4	583967.40	1583.23	28824.18	65852.07	124	6.51	11.2
2009	5000.0	28955.5	579670.40	1276.67	28484.12	34788.00	65	5.97	13.87
2010	5000.0	29776.3	576687.20	1909.71	28210.00	54964.38	117	5.86	12.25
2011	5000.0	31150.3	570990.80	1697.59	27800.14	48323.53	117	5.46	12.38
2012	5000.0	32962.2	566208.10	2285.16	27435.87	47893.41	127	5.18	8.31
2013	5000.0	34826.3	558901.80	594.41	26945.70	34136.72	150	5.27	8.0
2014	5000.0	36828.2	558844.80	1175.06	26817.71	39292.37	183	5.5	8.28
2015	5000.0	38898.5	556300.30	543.11	26565.45	31796.67	244	5.66	8.78
2016	5000.0	41060.9	556465.40	722.13	26448.75	32976.36	257	5.86	9.16
2017	5000.0	43297.8	555859.80	1177.43	26293.46	35893.86	258	6.02	9.45
2018	5000.0	45543.7	553309.70	836.77	26040.75	31058.64	179	6.12	9.93
2019	5000.0	47800.5	552215.80	624.20	25861.14	27783.48	234	6.27	9.95

Figura 5.1. Resultados da simulação apresentados pelo programa

Nessa tabela incluíram-se as variáveis fundamentais para uma análise expedita de uma simulação, descritas na tabela 5.1.

Tabela 5.1. Resultados da simulação - informação no ecrã

Nome	Unidade	Descrição
Ano	-	Ano da simulação
VolSC	m^3	Volume sem casca cortado - deve, na maior parte dos casos, ser idêntico ao volume de procura definido
Stock	m^3	Volume em pé antes da realização do corte
ArTot	ha	Área total ocupada com eucalipto
ArReconv	ha	Área que foi reconvertida
ArArdida	ha	Área consumida pelos fogos
AreaCort	ha	Área cortada
NAreaCort	-	Nº de áreas cortadas
IdMed	Anos	Idade média dos povoamentos em pé antes do corte
IdMedCrt	Anos	Idade média das áreas cortadas

5.2. Resultados em ficheiro

Os resultados totais de cada simulação são armazenados na directoria criada para o efeito durante a execução do programa. Cada simulação é descrita por um conjunto de 6 ficheiros.

5.2.1. Informação geral - geral.csv

O ficheiro geral.csv consiste num sumário executivo contém essencialmente a mesma informação apresentada na secção 5.1, ao que acresce o volume ardido (Tabela 5.2).

Tabela 5.2. Resultados da simulação - informação geral

Nome	Unidade	Descrição
Ano	-	Período da simulação
volcort	m^3	Volume sem casca cortado - deve, na maior parte dos casos, ser idêntico ao volume de procura definido
stock	m^3	Volume em pé antes da realização do corte
artot	ha	Área total ocupada com eucalipto
arreconv	ha	Área que foi reconvertida
arard	ha	Área consumida pelos fogos
volard	m^3	Volume ardido
arnova	ha	Áreas novas plantadas
arcort	ha	Área cortada
ncorts	-	Nº de áreas cortadas
idmed	anos	Idade média dos povoamentos em pé antes do corte
idmedcort	anos	Idade média das áreas cortadas

5.2.2. Simulação ao nível do concelho - concelhos.csv

Os resultados da simulação são também calculados ao nível do concelho incluindo para este casdo as áreas cortadas, abandonadas e reconvertidas, bem como o stock e o volume cortado (Tabela 5.3.)

Tabela 5.3. Resultados da simulação - Dados de simulação por concelho

Nome	Unidade	Descrição
Ano	-	Período da simulação
concelho	-	Código do concelho
arcort	ha	Área cortada
araband		Área total abandonada
arreconv	ha	Área que foi reconvertida
ncorts	-	Nº de áreas cortadas
stock	m^3	Volume em pé antes da realização do corte
volcort	m^3	Volume sem casca cortado no concelho
idmed	anos	idade média dos povoamentos em pé antes do corte no concelho
idmedcort	anos	Idade média das áreas cortadas no concelho

5.2.3. Simulação ao nível da área de gestão - sim_detalhe.csv

É possível seguir todos os povoamentos ao longo do período de simulação, incluindo áreas novas e povoamentos divididos (devido a fogos, reconversões ou cortes incompletos) através da análise do ficheiro *sim_detalhe.csv*.. Este ficheiro inclui ainda a informação biométrica de todas as áreas de gestão aou longo do horizonte de planeamento (tabela 5.4).

Tabela 5.4. Resultados da simulação - Dados de simulação por parcela

Nome	Unidade	Descrição
Ano	-	Período da simulação
stand_id	-	identificador interno do povoamento
concelho	-	Código do concelho
regiao	-	Área cortada
iqe	m	ìndice de qualidade da estação
area	ha	Área que foi reconvertida
rot	-	Nº de áreas cortadas
narvs	-	Nº de árvores maiores que 5 cm
hd	m	Altura dominante
ab	m2/ha	Área basal por ha
vol_sc	m3	Volume total sem casca

5.2.4. Áreas ardidas - fogos.csv

A tabela fogos.csv, contém a informação relevante à área ardida mostrada ao nível do concelho - áreas e volumes ardidos ao nível do concelho (Tabela 5.5.)

Tabela 5.5. Resultados da simulação - Áreas ardidas

Nome	Unidade	Descrição
Ano	-	Período da simulação
concelho	-	Código do concelho
arardida	ha	Área ardida por concelho
volardido	m^3	Volume ardido no concelho

5.2.5. Áreas cortadas - areas_cort.csv

A tabela de áreas cortadas apresenta um subconjunto da simulação ao nível da área de gestão, mostrando as áreas cortadas das diferentes unidades de gestão (tabela 5.6.)

Tabela 5.6. Resultados da simulação - Áreas cortadas

Nome	Unidade	Descrição
Ano	-	Período da simulação
stand_id	-	identificador interno do povoamento
regiao	-	Região CELPA
area	ha	volume cortado
iqe	m	ìndice de qualidade da estação
idade	anos	Idade de corte
volcort	m^3	Volume cortado

5.2.6. Áreas novas - areas_novas.csv

As áreas novas a plantar são repartidas pelos concelhos presentes nas diferentes áreas de gestão de acordo com a percentagem de ocupação de eucaliptal verificada. A tabela 5.7 apresenta qual a área nova instalada anualmente por concelho

Tabela 5.7. Resultados da simulação - Áreas novas plantadas

Nome	Unidade	Descrição
Ano	-	Período da simulação
concelho	-	Código do concelho
arnova	ha	Área nova plantada no concelho

Agradecimentos:

Parte do financiamento que deu origem a este projecto foi concedido ao abrigo do Projecto PTDC/AGR-CFL/64146/2006

Apêndice - Código fonte do programa CELPASIM 1.0.0

Dos 15 módulos que constituem o programa CELPASIM apresenta-se o código fonte de 3:

A. O módulo de simulação do crescimento, a que respeita todo o código relativo aos procedimentos de simulação e as equações biométricas utilizadas.(globais.bas)

B. O módulo de gestão da aplicação, que controla todo o desenrolar do processo de simulação (other.bas)

C. O módulo de input-output, onde estão definidos os procedimentos de leitura de dados e escrita de resultados (io.bas)

A.1. Módulo globais.bas

```
Option Explicit
Const LINEAR = 1
Const NAO_LINEAR = 0
Public SIM_PATH As String
Public NREGIONS As Long
Public ANO_REFERENCIA As Long
Public NCONCELHOS As Long
Public NStands As Long
Public NYEARS As Long
Public PROB_ABANDONO As Single
Public PROB_RECONVERTE As Single
Public NVARAS_TOICA As Single
```

```
Public DATABASE_NAME As String

Type ARegion
    iqe As Single
    narvs_plant As Single
    NCONCELHOS As Long      'no de concelhos na regiao
    Concelhos() As Long     'código dos concelhos
    prop_area() As Single   'proporção da área de eucaliptal em cada um (Sum =
1.0)
End Type

Type AConcelho
    regiao As Long
    area As Single
End Type

Public Type Result
    st_harv As Long
    area_em_pe As Single
    area_nova As Single
    area_cort As Single
    area_aband As Single
    area_reconv As Single
    area_ardida As Single
    stock As Single
    idade_med As Single
    vol_cort As Single
    vol_ardido As Single
    idade_med_cort As Single
End Type

Public Type Resultado
    Regioes() As Result
    Concelhos() As Result
End Type


Public Resultados() As Resultado
Public Regions() As ARegion
Public Stands() As Stand
Public HStands() As HStand
Public NewAreas() As Single
Public BurntConcelhos() As AConcelho
'Public Model_Type As Long
Public VolTargetSC() As Single    'volume targets for each year
Public BurntAreas() As Single        'proportion of burn area every year


Public Function calc_idade(ab As Single, hdom As Single, n_M5 As Single,
age_est_low)
    Dim a As Single, b As Single, c As Single, d As Single, e As Single
    a = 4.15124
    b = 0.05142
    c = 0.04318
    d = -0.00101
    e = 0.67179
    calc_idade = a + b * ab + c * hdom + d * n_M5 + e * age_est_low
End Function

Public Function calc_hdom_lin(hdom1 As Single, ab1 As Single, n_M5 As Single)
    Dim a As Single, b As Single, c As Single, d As Single, e As Single
    a = 7.05581
    b = 0.16797
    c = 1.4096
    d = 0.1265
    e = 41.30148
    calc_hdom_lin = a + b * hdom1 ^ c + d * ab1 + e * ab1 / n_M5
End Function

Public Function calc_abas_lin(hdom1 As Single, ab1 As Single, FW As Single)
    Dim a As Single, b As Single, c As Single, d As Single, e As Single
    a = 2.93909
    b = 0.94937
    c = 1.0194
```

```
    d = -0.05523
    e = -4.18516
    calc_abas_lin = a + b * ab1 ^ c - d * hdom1 - e * FW
End Function

Public Function calc_voltot_lin(st As Stand)
    'vol com toiça e com casca
    Dim a As Single, b As Single, c As Single, fe As Single, kvc As Single
    Dim rot As Long, r As Long
    If st.age_known = False Then
        a = 0.5207
        b = 0.9933
        c = 0.9285
        calc_voltot_lin = a * st.hd ^ b * st.ba ^ c
     Else
        rot = st.rot
        a = 0.0655
        b = 0.883
        c = 1.0263
        fe = 100 / (st.iqe * Sqr(st.narvs_plant))
        kvc = 0.4886 - 0.0387 * (rot > 1) - 0.0122 * (r = 1) + 0.248 * fe *
(rot > 1) + 0.1348 * fe * (r = 1) - 0.0039 * (r = 1) * (rot > 1)
        calc_voltot_lin = kvc * st.age ^ a + st.hd ^ b + st.ba ^ c
    End If

End Function

Public Function calc_volctsc_lin(st As Stand)
    'vol com toiça, sem casca
    Dim a As Single, b As Single, c As Single, fe As Single, kvs As Single
    Dim rot As Long, r As Long

    If st.age_known = False Then
        a = 0.3625
        b = 0.9885
        c = 0.9879
        calc_volctsc_lin = a * st.hd ^ b * st.ba ^ c
       Else
        a = 0.0592
        b = 0.09349
        c = 1.0077
        rot = st.rot
        r = ConcCodes(st.concelho).reg_glob
        fe = 100 / (st.iqe * Sqr(st.narvs_plant))
        kvs = 0.3724 - 0.0384 * (rot > 1) - 0.0162 * (r = 1) + 0.2119 * fe *
(rot > 1) + 0.1497 * fe * (r = 1) - 0.0052 * (r = 1) * (rot > 1)
        calc_volctsc_lin = kvs * st.age ^ a + st.hd ^ b + st.ba ^ c
    End If
End Function

Public Function calc_volstcc_lin(hdom As Single, ab As Single, rot As Long)
   'vol sem toiça com casca
    Dim a As Single, a1 As Single, b As Single, c As Single
    a = 0.4602
    a1 = 0.0012
    b = 0.9796
    c = 0.975
    calc_volstcc_lin = (a + a1 * rot) * hdom ^ b * ab ^ c
End Function

Public Function calc_volstsc_lin(st As Stand, hdom As Single, ab As Single, rot
As Long) As Single
   'vol sem toiça sem casca
    Dim a As Single, a1 As Single, b As Single, c As Single, rota As Long
    Dim fe As Single, kvf As Single, kv0 As Single, kv As Single
    Dim r As Long
    If rot = 1 Then rota = 1 Else rota = 0
    If st.age_known = False Then
        a = 0.3202
        a1 = 0.00231
        b = 0.9756
        c = 1.0343
        calc_volstsc_lin = (a + a1 * rota) * hdom ^ b * ab ^ c
       Else
```

```
        If st.age > 8 Then
            a = a
        End If
        a = 0.0592
        b = 0.9349
        c = 1.0077
        r = ConcCodes(st.concelho).reg_glob
        Select Case r
            Case 1, 2, 3, 6
                If rot = 1 Then
                    kv0 = 0.3886
                    kvf = -0.1497
                Else
                    kv0 = 0.4218
                    kvf = -0.3616
                End If
            Case 4, 5, 8
                If rot = 1 Then
                    kv0 = 0.3724
                    kvf = 0
                Else
                    kv0 = 0.4108
                    kvf = -0.2119
                End If
        End Select
        fe = 100 / (st.iqe * Sqr(st.narvs_plant))
        kv = kv0 + kvf * fe
        calc_volstsc_lin = kv * st.age ^ a * st.hd ^ b * st.ba ^ c
    End If


End Function

Public Function calc_volmercantil_cc(volstcc As Single, dmed As Single, ddesp
As Single)
    Dim a As Single, b As Single
    a = -1.0033
    b = 3.8608
    calc_volmercantil_cc = volstcc * Exp(a * (ddesp / dmed) ^ b)
End Function

Public Function calc_volmercantil_sc(volstcc As Single, dmed As Single, ddesp
As Single)
    Dim a As Single, b As Single
    a = -1.0173
    b = 3.8929
    calc_volmercantil_sc = volstcc * Exp(a * (ddesp / dmed) ^ b)
End Function

Public Function calc_narvs_n(st As Stand)
    'calcula o número de árvores com base no narvs anterior
    Dim am  As Single
    Dim rot As Long, r As Long
    rot = st.rot
    r = ConcCodes(st.concelho).reg_glob
    If st.age_known = False Then
        calc_narvs_n = st.narvs_init * Exp(-0.00988 * (st.age - st.age_init))
      Else
        am = 0.0401 + 0.0013 * st.narvs_plant / 1000 - 0.0064 * st.iqe / 10 +
0.019 * ((r = 1) + (r = 7)) + 0.0159 * (rot = 5) - _
            (-0.0311 - 0.0152 * ((r = 1) + (r = 5) + (r = 7))) * (rot > 1) -
(0.005 * st.iqe / 10) * (rot > 1)
        If st.rot = 1 Then
            calc_narvs_n = st.narvs_plant * Exp(-am * st.age)
         Else
        'atenção ao n_arvs_init!!!!!!!!!
            calc_narvs_n = st.narvs_plant * NVARAS_TOICA * Exp(-am * (st.age -
3))
        End If
    End If
End Function

Public Function calc_narvs_nM5_lin(narvs2 As Single, ab2 As Single, id2 As
Single)
```

```
    'calcula o número de árvores  maiores que 5 cm
    Dim a As Single, b As Single, c As Single
    a = -10.52037
    b = 0.85492
    c = 6.00689
    calc_narvs_nM5_lin = a + b * narvs2 + c * ab2
End Function

Public Function calc_narvs_nM5(narvs2 As Single, hdom2 As Single)
    'calcula o número de árvores  maiores que 5 cm
    Dim a As Single, b As Single, c As Single
    a = -1.5163
    b = 0.1075
    c = 1 / 6#
    calc_narvs_nM5 = narvs2 / (1 + Exp(-(a + b * hdom2))) ^ c
End Function

Public Function calc_hdom(st As Stand, ab1 As Single, hd1 As Single)
    Dim a As Single, k As Single, n As Single, r As Long
    'O k tem de ser calculado antes. Se for negativo corre o globulus (patch de
9/6/05)
    k = 3.0839 - 0.1142 * ab1 + 0.1202 * hd1
    If st.age_known = False And k > 0 Then
        a = 84.2463
        n = 0.4057
        calc_hdom = a * Exp(-k / ((-k / Log(hd1 / a)) ^ (1 / n) + 1) ^ n)
        Exit Function
     Else
        If st.age = 0 Then
            calc_hdom = 0
            Exit Function
        End If
        a = 61.1372
        r = ConcCodes(st.concelho).reg_glob
        Select Case r
            Case 1, 2
                If st.rot = 1 Then n = 0.5225 Else n = 0.4384
            Case 3
                If st.rot = 1 Then n = 0.4805 Else n = 0.3964
            Case 4
                If st.rot = 1 Then n = 0.4407 Else n = 0.2826
            Case 5
                If st.rot = 1 Then n = 0.478 Else n = 0.3199
            Case 6
                If st.rot = 1 Then n = 0.4805 Else n = 0.3964
            Case Else
                If st.rot = 1 Then n = 0.3955 Else n = 0.2374
        End Select
        calc_hdom = a * (st.iqe / a) ^ ((10 / st.age) ^ n)
    End If

End Function

Public Function calc_abas(st As Stand, ab1 As Single, hd1 As Single, nm51 As
Single)
    Dim a As Single, k As Single, n As Single
    Dim kgq As Single, kgnp As Single, kgf As Single, kg0 As Single
    Dim fe As Single, kg As Single, ng0 As Single, ngq As Single, ngn As Single
    Dim ag As Single, ng As Single, ab2 As Single, m As Single
    Dim r As Long
    If Not st.age_known Then
        a = 63.9939 * hd1
        k = 9.8573 + 0.442 * hd1 - 2.289 * nm51 / 1000
        n = 0.4141
        ab2 = a * Exp(-k / ((-k / Log(ab1 / a)) ^ (1 / n) + 1) ^ n)
        ' entrada na função de socorro se o acréscimo for superior a 3m2 ou a
95% do valor da assímptota
        ' (grande martelada!) ---22/2/2005
        If ab2 - ab1 > 3 Or ab2 >= 0.95 * 63.81254 Then
            m = 63.81254
            a = 1.381175
            ab2 = m / (1 - (1 - m / ab1) * (st.age / (st.age + 1)) ^ a)
        End If
        calc_abas = ab2
```

```
   Exit Function
Else
   If st.age = 0 Then
       calc_abas = 0
       Exit Function
   End If
   Select Case ConcCodes(st.concelho).reg_glob
       Case 1, 2
           If st.rot = 1 Then
               kg0 = -4.482
               ng0 = 3.735
           Else
               kg0 = 0.7582
               ng0 = 2.8638
           End If
       Case 3
           If st.rot = 1 Then
               kg0 = -5.5311
               ng0 = 3.6354
           Else
               kg0 = -0.2903
               ng0 = 2.7642
           End If
       Case 4
           If st.rot = 1 Then
               kg0 = -6.1201
               ng0 = 3.6091
           Else
               kg0 = -0.8793
               ng0 = 2.699
           End If
       Case 5
           If st.rot = 1 Then
               kg0 = -6.2433
               ng0 = 3.5676
           Else
               kg0 = -1.0025
               ng0 = 2.6575
           End If
       Case 6
           If st.rot = 1 Then
               kg0 = -4.756
               ng0 = 3.6893
           Else
               kg0 = 0.4848
               ng0 = 2.8181
           End If
       Case 7
           If st.rot = 1 Then
               kg0 = -7.7941
               ng0 = 3.4571
           Else
               kg0 = -2.5533
               ng0 = 2.547
           End If
       Case 8
           If st.rot = 1 Then
               kg0 = -6.9326
               ng0 = 3.5676
           Else
               kg0 = -1.6918
               ng0 = 2.6575        '!!!!!!ESte valor não está lá!!!!!!!
           End If
   End Select
   If st.rot = 1 Then
       kgq = 177.9
       kgnp = 0.5408
       kgf = 16.015
    Else
       kgq = 72.3
       kgnp = 0.0134
       kgf = 14.1898
   End If
   ag = 0.1586
```

```
        fe = 100 / (st.iqe * Sqr(st.narvs_plant))
        kg = kg0 + kgq / st.iqe + kgnp * st.narvs_plant / 1000 + kgf * fe
        r = ConcCodes(st.concelho).reg_glob
        'ng0 = 3.4571 - 0.232 * (r = 1) + 0.0539 * (r = 2) - 0.0996 * (r = 3) -
0.1105 * (r = 5) - 0.152 * (r = 7)
        'ng0 = ng0 - (-0.9101 - 0.0389 * (r = 1)) * (st.rot > 1)
        ngq = -1.0288 - 0.2972 * (st.rot > 1)
        'ngn = 0.1024 + 0.8008 * (st.rot > 1)
        If st.rot = 1 Then ngn = 0.1024 Else ngn = 0.0216
        ng = ng0 + ngq * Log(st.iqe) + ngn * st.narvs_plant / 1000
        calc_abas = ag * st.iqe * st.iqe * Exp(-kg * (1 / st.age) ^ ng)
        ' calc_abas=
    End If
End Function



Public Function simulate(st As Stand)
    Dim ab1 As Single, hd1 As Single
    Dim nm51 As Single
    If st.harvested = True Then
        st.age = 1
        st.ba = 0
        st.hd = 0
        st.narvs = 0
        st.vol_cc = 0
        st.vol_sc = 0
    End If
    ab1 = st.ba
    hd1 = st.hd
    nm51 = st.nm5
    st.ba = calc_abas(st, ab1, hd1, nm51)
    st.hd = calc_hdom(st, ab1, hd1)
    st.narvs = calc_narvs_n(st)
    st.nm5 = calc_narvs_nM5(st.narvs, st.hd)
    'atenção!!!!!
    If st.nm5 = 0 Then
        st.dqm = 0
       Else
        If st.age_known = False Then
            st.dqm = 200 * Sqr((st.ba / st.nm5) / 3.1416)
           Else
            st.dqm = 200 * Sqr((st.ba / st.narvs) / 3.1416)
        End If
    End If
    st.vol_sc = calc_volstsc_lin(st, st.hd, st.ba, st.rot)
    st.vol_cc = calc_voltot_lin(st)
    'st.vol_sc = calc_volctsc_lin(st)
    'st.vol_cc = calc_volctsc_lin(st)

End Function



Public Sub clearcut(year As Long, st As Stand)
    ' here we clearcut a full stand!
    ' we must account for the stand harvested
    ' and fill in the harvested volume
    Dim st_harv As Long
    Dim res As Result
    If st.id = "246" Then
        st_harv = st_harv
    End If
    res = Resultados(year).Concelhos(st.concelho)
    'we must save the harvested stand
    st_harv = HStands(year).st_harv + 1
    res.st_harv = res.st_harv + 1
    ReDim Preserve HStands(year).Stands(st_harv)
    Call StandCopy(HStands(year).Stands(st_harv), st, 1)

    HStands(year).st_harv = st_harv
    'update the volumes
```

```
    HStands(year).vol_sc = HStands(year).vol_sc + st.vol_sc * st.area
    HStands(year).vol_cc = HStands(year).vol_cc + st.vol_cc * st.area
    HStands(year).area_cort = HStands(year).area_cort + st.area
    res.vol_cort = res.vol_cort + st.vol_sc * st.area
    res.area_cort = res.area_cort + st.area

    '1a parte do cálculo da idade média de árvores cortadas
    res.idade_med_cort = res.idade_med_cort + st.age * st.area
    HStands(year).idade_med_corte = HStands(year).idade_med_corte + st.age *
st.area
    'abandon happens when the rotation is larger than 4 and is performed by
reducing
    'the area of the stand
    If st.rot >= 3 And PROB_ABANDONO > 0 Then '!!!!!!!!
        HStands(year).area_aband = HStands(year).area_aband + st.area *
PROB_ABANDONO
        res.area_aband = res.area_aband + st.area * PROB_ABANDONO
        st.area = st.area * (1 - PROB_ABANDONO)
    End If
    'update the stand data
    'st.age = 0
    st.ba = 0
    st.hd = 0
    st.narvs = 0 '!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
    st.vol_cc = 0
    'st.vol_sc = 0
    st.harvested = True
    st.age_known = True
    'update the number of trees if we have a clearcut in the first rotation
    If st.rot = 1 Then
        st.narvs_plant = st.narvs_plant * NVARAS_TOICA
    End If
    st.rot = st.rot + 1
    Resultados(year).Concelhos(st.concelho) = res
End Sub


Public Function Grow()
    ' grows all stands to the next year
    ' note that some areas may get to be abandoned
    Dim s As Long
    For s = 1 To NStands
        '
        ' here is the flag - only stands with rot >=0 will be simulated
(23/5/05)
        '
        If Stands(s).rot > 0 Then
            Stands(s).age = Stands(s).age + 1
            Call simulate(Stands(s))
        End If
        Stands(s).harvested = False
    Next
End Function
```

A.2. Módulo other.bas

```
Option Explicit
'stand codes
Public Const OK = 0
Public Const QUEIMADO = 1
Public Const ABANDONADO = 2
Public Const MUITO_JOVEM = 3
Public NomeRegiao() As String

Public Type ConcCode
    nome As String
    cod As Long
    reg_glob As Long 'Região do Globulus
End Type

Public ConcCodes() As ConcCode
Public NConcCodes As Long
```

```
Public Type Ranking
    nclasses As Long
    maxclass() As Single
    valor() As Single
End Type

Public RankIdade As Ranking
Public RankDQM As Ranking
Public RankRegiao() As Single

Public Type Stand
    rank As Long
    cod As Long
    id As String
    name As String
    region As Long
    area As Single
    concelho As Long
    age As Single
    rot As Long
    iqe As Single
    age_init As Single
    age_known As Boolean
    dqm As Single
    ba_init As Single
    hd_init As Single
    narvs_init As Single
    narvs_plant As Single     'initialized only after a clearcut
    nm5_init As Single
    ba As Single
    hd As Single
    narvs As Single
    nm5 As Single               'n trees larger than 5 cm
    vol_sc As Single
    vol_cc As Single
    harvested As Boolean
End Type

Public Type HStand
    st_harv As Long
    vol_cc As Single
    vol_sc As Single
    stock As Single
    area_nova As Single
    area_aband As Single
    area_ardida As Single
    area_reconv As Single
    area_cort As Single
    area_em_pe As Single
    volume_ardido As Single
    idade_med As Single
    idade_med_corte As Single
    Stands() As Stand
End Type



Public Function RRegiao(regiao As Long) As Single
    RRegiao = RankRegiao(regiao)
End Function
Public Function RIdade(idade As Single) As Long
    Dim r As Long
    For r = RankIdade.nclasses To 1 Step -1
        If idade >= RankIdade.maxclass(r) Then
            RIdade = RankIdade.valor(r)
            Exit Function
        End If
    Next
    Exit Function
    Select Case Int(idade)
        Case 6 To 7
            RIdade = 0
        Case 8 To 9
            RIdade = 4
```

```
        Case 10 To 11
            RIdade = 6
        Case 12 To 13
            RIdade = 3
        Case 14 To 15
            RIdade = 1
        Case Is >= 16
            RIdade = 0
    End Select
End Function

Public Function RDQM(dqm As Single) As Long
'ranking criteria for each dqm value
    Dim r As Long
    For r = RankDQM.nclasses To 1 Step -1
        If dqm >= RankDQM.maxclass(r) Then
            RDQM = RankDQM.valor(r)
            Exit Function
        End If
    Next

    Select Case dqm
        Case 7.5 To 10
            RDQM = 1
        Case 10 To 12.5
            RDQM = 2
        Case 12.5 To 15
            RDQM = 3
        Case 15 To 17.5
            RDQM = 4
        Case Is >= 17.5
            RDQM = 5
    End Select

End Function


Private Function Compare(stand1 As Stand, stand2 As Stand)
    ' this is where each stand is compared to the other one
    ' it is defined by the runtimes
    ' if stand1 is LOWER than stand2 returns -1 Else returns 0
    Dim rank1 As Long, rank2 As Long
    rank1 = RRegiao(stand1.region)
    rank1 = rank1 + RIdade(stand1.age)
    rank1 = rank1 + RDQM(stand1.dqm)
    rank2 = RRegiao(stand2.region)
    rank2 = rank2 + RIdade(stand2.age)
    rank2 = rank2 + RDQM(stand2.dqm)
    stand1.rank = rank1
    stand2.rank = rank2

    If rank1 = rank2 Then 'desempata-se com o volume
        rank1 = rank1 - (stand1.vol_sc > stand2.vol_sc)
    End If
    ' the rot flag again - these stands should never be harvested!!! (23/06/05)
    If stand1.rot = 0 Then rank1 = rank1 - 1000
    If stand2.rot = 0 Then rank2 = rank2 - 1000
    Compare = rank1 > rank2 'order descending
End Function


Public Sub QSort(low As Long, high As Long)
   ' basic qsrot routine coirtesyy of randy birch. Thanks!
   Dim pivot   As Stand
   Dim tmpSwap As Stand
   Dim tmpLow  As Long
   Dim tmpHigh   As Long

   tmpLow = low
   tmpHigh = high
   pivot = Stands(Int((low + high) / 2))
   While (tmpLow <= tmpHigh)
      ' compare 1
      While Compare(Stands(tmpLow), pivot) And tmpLow < high
```

```
            tmpLow = tmpLow + 1
         Wend
         'compare 2
         While Compare(pivot, Stands(tmpHigh)) And tmpHigh > low
            tmpHigh = tmpHigh - 1
         Wend
         'swap
         If (tmpLow <= tmpHigh) Then
            tmpSwap = Stands(tmpLow)
            Stands(tmpLow) = Stands(tmpHigh)
            Stands(tmpHigh) = tmpSwap
            tmpLow = tmpLow + 1
            tmpHigh = tmpHigh - 1
         End If
      Wend
      If (low < tmpHigh) Then Call QSort(low, tmpHigh)
      If (tmpLow < high) Then Call QSort(tmpLow, high)
End Sub

Public Sub StandCopy(s_to As Stand, s_from As Stand, ratio As Single)
' copies s_from to s_to, multiplying the area by ratio
    s_to.id = s_from.id
    s_to.name = s_from.name
    s_to.area = s_from.area * ratio
    s_to.region = s_from.region
    s_to.concelho = s_from.concelho
    s_to.age = s_from.age
    s_to.age_known = s_from.age_known
    s_to.dqm = s_from.dqm
    s_to.rot = s_from.rot
    s_to.iqe = s_from.iqe
    s_to.ba_init = s_from.ba_init
    s_to.hd_init = s_from.hd_init
    s_to.ba = s_from.ba
    s_to.hd = s_from.hd
    s_to.vol_sc = s_from.vol_sc
    s_to.vol_cc = s_from.vol_cc
    s_to.narvs_plant = s_from.narvs_plant
End Sub

Public Sub Harvest(year As Long)
' this is a tricky routine as it involves many things
' 1: accomplish the objective
' 2: create, if necessary, new stands
    Dim s As Long, c As Long, h As Single
    Dim vol As Single, voltot As Single
    Dim volneeded As Single, ratio As Single
    voltot = 0
    s = 1
    h = 0
    If VolTargetSC(year) = 0 Then
        Exit Sub
    End If
    While s <= NStands
        vol = Stands(s).vol_sc * Stands(s).area
        If voltot + vol <= VolTargetSC(year) Then
            Call clearcut(year, Stands(s))
            voltot = voltot + vol
            h = h + Stands(s).vol_sc * Stands(s).area
         Else
            ' first determine the ratio of the area to harvest
            volneeded = VolTargetSC(year) - voltot
            ratio = volneeded / vol
            ' now create a new stand
            NStands = NStands + 1
            ReDim Preserve Stands(NStands)
            'copy the apropriate area to the new place
            Call  StandCopy(Stands(NStands),  Stands(s),  ratio)  '  copies  the
original data to the new one
            Stands(NStands).id = "CT_" & Stands(s).id & "_" & Trim(Str(year))
            Stands(s).area = Stands(s).area * (1 - ratio)
            'and bang! harvest it!
            Call clearcut(year, Stands(NStands))
            h = h + Stands(NStands).vol_sc * Stands(NStands).area
```

```
            Debug.Print year, h
            Exit Sub
        End If

        s = s + 1
    Wend
    Debug.Print year, h

End Sub

Private Sub BurnForest(year As Long)
' acumula as áreas queimadas por concelho e, caso a área seja relevante
' cria povoamentos novos!
    Dim s As Long, c As Long, conc As Long, reg As Long
    Dim res As Result, ta As Single, ba As Single
    'limpar as áreas
    year = year
    For c = 1 To NCONCELHOS
        BurntConcelhos(c).area = 0
    Next
    'preencher as áreas ardidas acumulando nos concelhos
    ba = 0
    For s = 1 To NStands
        reg = Stands(s).region
        res = Resultados(year).Concelhos(Stands(s).concelho)
        If BurntAreas(reg) > 0 Then
            conc = Stands(s).concelho
            BurntConcelhos(conc).area     =     BurntConcelhos(conc).area     +
Stands(s).area * BurntAreas(reg)
            ba = ba + Stands(s).area * BurntAreas(reg)
            HStands(year).area_ardida     =     HStands(year).area_ardida     +
Stands(s).area * BurntAreas(reg)
            'HStands(year).volume_ardido    =    HStands(year).volume_ardido    +
Stands(s).area * Stands(NStands).vol_sc * BurntAreas(reg)
            HStands(year).volume_ardido    =    HStands(year).volume_ardido    +
Stands(s).area * Stands(s).vol_sc * BurntAreas(reg)
            res.area_ardida    =    res.area_ardida    +    Stands(s).area    *
BurntAreas(reg)
            'res.vol_ardido    =    res.vol_ardido    +    Stands(s).area    *
Stands(NStands).vol_sc * BurntAreas(reg)
            res.vol_ardido = res.vol_ardido + Stands(s).area * Stands(s).vol_sc
* BurntAreas(reg)
            Stands(s).area = Stands(s).area * (1 - BurntAreas(reg))
        End If
        Resultados(year).Concelhos(Stands(s).concelho) = res
    Next
    ta = 0
    For c = 1 To NCONCELHOS
        'só crio áreas novas se a área ardida no concelho for superior a 10 ha
        If BurntConcelhos(c).area > 10 Then
            NStands = NStands + 1
            ReDim Preserve Stands(NStands)
            Stands(NStands).concelho = c
            Stands(NStands).region = BurntConcelhos(c).regiao
            Stands(NStands).area = BurntConcelhos(c).area
            ta = ta + BurntConcelhos(c).area
            Stands(NStands).rot = 2 '!!!!!!!!!!!!!!!
            Stands(NStands).age = 0
            Stands(NStands).hd = 0
            Stands(NStands).ba = 0
            Stands(NStands).vol_sc = 0
            Stands(NStands).vol_cc = 0
            Stands(NStands).iqe = Regions(BurntConcelhos(c).regiao).iqe
            Stands(NStands).narvs_plant                                        =
Regions(BurntConcelhos(c).regiao).narvs_plant
            Stands(NStands).age_known = True
            Stands(NStands).id = "FG_" & Trim(Str(year)) & "_" & Trim(Str(c))
        End If
    Next
    'Debug.Print ba, ta
End Sub
Public Sub CalcStock(year As Long)
    Dim s As Long, c As Long
    For s = 1 To NStands
```

```
        HStands(year).stock   =   HStands(year).stock   +   Stands(s).vol_sc   *
Stands(s).area
        Resultados(year).Concelhos(Stands(s).concelho).stock                    =
Resultados(year).Concelhos(Stands(s).concelho).stock   +   Stands(s).vol_sc   *
Stands(s).area / 1000
        'para calcular a idade média multiplica-se a idade pela área
        'If Stands(s).age > 0 Then
            HStands(year).area_em_pe      =      HStands(year).area_em_pe      +
Stands(s).area
            HStands(year).idade_med = HStands(year).idade_med + Stands(s).area
* Stands(s).age
            Resultados(year).Concelhos(Stands(s).concelho).area_em_pe          =
Resultados(year).Concelhos(Stands(s).concelho).area_em_pe + Stands(s).area
            Resultados(year).Concelhos(Stands(s).concelho).idade_med           =
Resultados(year).Concelhos(Stands(s).concelho).idade_med   +   Stands(s).area   *
Stands(s).age
        'End If
    Next
    ' e agora sim. calcular a verdadeira idade média!
    For c = 1 To NCONCELHOS
        If Resultados(year).Concelhos(c).area_em_pe > 0 Then
            Resultados(year).Concelhos(c).idade_med                            =
Resultados(year).Concelhos(c).idade_med                                        /
Resultados(year).Concelhos(c).area_em_pe
        End If
        ' e a idade média de corte!
        If Resultados(year).Concelhos(c).area_cort > 0 Then
            Resultados(year).Concelhos(c).idade_med_cort                       =
Resultados(year).Concelhos(c).idade_med_cort                                   /
Resultados(year).Concelhos(c).area_cort
        End If
    Next
    If HStands(year).area_em_pe > 0 Then
        HStands(year).idade_med        =        HStands(year).idade_med        /
HStands(year).area_em_pe
    End If
    'If HStands(year).area_cort > 0 Then
    '        HStands(year).idade_med_corte  =  HStands(year).idade_med_corte  /
HStands(year).area_cort
    'End If
End Sub

Public Sub Abandon(year As Long)
    'abandon happens when the rotation is larger than 4 and is performed by
reducing
    'the area of the stand
    Dim s As Long
    Dim st As Stand
    Dim res As Result
    For s = 1 To NStands
        st = Stands(s)
        res = Resultados(year).Concelhos(st.concelho)
        If  st.harvested  =  True  And  st.rot  >=  4  And  PROB_ABANDONO  >  0  Then
'!!!!!!!!
            HStands(year).area_aband  =  HStands(year).area_aband  +  st.area  *
PROB_ABANDONO
            res.area_aband = res.area_aband + st.area * PROB_ABANDONO
            st.area = st.area * (1 - PROB_ABANDONO)
        End If
        Resultados(year).Concelhos(st.concelho) = res
    Next
End Sub

Public Sub RunSimulation(years As Long)
    ' the very basic and simple routine that coordinates everything

    Dim y As Long, i As Long, s As Long, st As Stand, c As Long
    Dim res As Result, atot As Single
    'Open "teste.csv" For Output As #10

    Open SIM_PATH & "\sim_detalhe.csv" For Output As #113
    Print #113, "ano, stand_id, concelho, regiao, iqe, area, rot, idade, narvs,
hd, ab, vol_sc"
    For y = 1 To years
```

```
        Call Grow
        Call BurnForest(y)
        Call QSort(1, NStands)
        Call CalcStock(y)
        Call Harvest(y)
        Call Reconverte(y)
        'Call Abandon(y)
        Call AddNewAreas(y)
        If HStands(y).area_cort > 0 Then
            HStands(y).idade_med_corte    =    HStands(y).idade_med_corte    /
HStands(y).area_cort
        End If
        For s = 1 To NStands
            Write #113, y, Stands(s).id, Stands(s).concelho, Stands(s).region,
Stands(s).iqe,       Stands(s).area,       Stands(s).rot,       Stands(s).age,
CInt(Stands(s).narvs), Stands(s).hd, Stands(s).ba, Stands(s).vol_sc
        Next
        'Debug.Print     y,     HStands(y).vol_sc,     HStands(y).area_reconv,
HStands(y).area_ardida,        HStands(y).area_cort,        HStands(y).st_harv,
HStands(y).stock
    Next
    Close #113
    Call write_results
End Sub

Public Sub AddNewAreas(year As Long)
    'adds areas, according to the requirements, for each region
    Dim r As Long, c As Long, are_tot As Single, pa As Single
    For r = 1 To NREGIONS
        If NewAreas(r, year) > 0 Then
            pa = 0
            For c = 1 To Regions(r).NCONCELHOS
                If NewAreas(r, year) * Regions(r).prop_area(c) > 0 Then
                    NStands = NStands + 1
                    ReDim Preserve Stands(NStands)
                    Stands(NStands).area     =     NewAreas(r,     year)     *
Regions(r).prop_area(c)
                    pa = pa + Regions(r).prop_area(c)
                    are_tot = are_tot + Stands(NStands).area
                    Stands(NStands).concelho = Regions(r).Concelhos(c)
                    Stands(NStands).age = 0
                    Stands(NStands).rot = 1
                    Stands(NStands).id  =  "NA_"  &  Trim(Str(year))  &  "_"  &
Trim(Str(c))
                    Stands(NStands).iqe = Regions(r).iqe
                    Stands(NStands).age_known = True
                    Stands(NStands).narvs_plant = Regions(r).narvs_plant

Resultados(year).Concelhos(Stands(NStands).concelho).area_nova                =
Stands(NStands).area
                End If
            Next
            'Debug.Print r, Regions(r).NCONCELHOS, pa
        End If
        HStands(year).area_nova = HStands(year).area_nova + NewAreas(r, year)
    Next
End Sub

Public Function Reconverte(year As Long)

    'handle reconversions
    Dim res As Result, st As Stand, s As Long, NS As Long, atot As Single, t As
Long
    NS = NStands
    For s = 1 To NS
        st = Stands(s)
        res = Resultados(year).Concelhos(st.concelho)
        'reconversões são áreas que acabaram de ser cortadas (idade=0) e têm
rot>3
        If st.rot > 3 And st.harvested = True And PROB_RECONVERTE > 0 Then
'!!!!!!!!!
'        If st.rot > 3 And st.age = 0 And PROB_RECONVERTE > 0 Then '!!!!!!!!!
            HStands(year).area_reconv = HStands(year).area_reconv + st.area *
PROB_RECONVERTE
```

```
                res.area_reconv = res.area_reconv + st.area * PROB_RECONVERTE
                If PROB_RECONVERTE < 0.99999 Then
                    NStands = NStands + 1
                    ReDim Preserve Stands(NStands)
                    Call StandCopy(Stands(NStands), Stands(s), PROB_RECONVERTE)
                    Stands(NStands).iqe    =    Stands(NStands).iqe    +    2    /
        (Stands(NStands).iqe / Regions(st.region).iqe)
                    Stands(NStands).rot = 1
                    Stands(NStands).narvs_plant = Regions(st.region).narvs_plant
                    Stands(NStands).id = "REC_" & st.id & "_" & Trim(Str(year))
                    Stands(s).area = Stands(s).area * (1 - PROB_RECONVERTE)
                    Stands(NStands).age = 0
                    Stands(NStands).ba = 0
                    Stands(NStands).hd = 0
                    Stands(NStands).narvs = 0 '!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
                    Stands(NStands).vol_cc = 0
                    Stands(NStands).vol_sc = 0
                    'O  IQE  depois  da  reconversão  é  igual  ao  IQE  antes  da
        reconversão + 2 /
                    '(IQE antes da reconversão / IQE médio região (constante no
        quadro)) FG
                Else
                    st.iqe  =  Stands(NStands).iqe  +  2  /  (Stands(NStands).iqe  /
        Regions(st.region).iqe)
                    st.rot = 1
                    st.narvs_plant = Regions(st.region).narvs_plant
                    st.age = 0
                    st.ba = 0
                    st.hd = 0
                    st.narvs = 0
                    st.vol_cc = 0
                    st.vol_sc = 0
                End If
            End If
            Resultados(year).Concelhos(st.concelho) = res
        Next
    End Function
```

A.3. Módulo de entrada e saída

```
Option Explicit

Public Sub init_data()
    Dim y As Long, r As Long
    ChDir App.Path
    ReDim VolTargetSC(NYEARS)
    ReDim HStands(NYEARS)
    ReDim NewAreas(NREGIONS, NYEARS)
    ReDim BurntAreas(NREGIONS)
    ReDim Regions(NREGIONS)
    ReDim RankRegiao(NREGIONS)
    ReDim Resultados(NYEARS)
    Call read_iqes("iqes.csv")
    Call read_narvs_plant("n_plants.csv")
    Call read_concelhos("areas_conc.csv")
    For y = 1 To NYEARS
        VolTargetSC(y) = 1000000
        For r = 1 To NREGIONS
            NewAreas(r, y) = 500
            BurntAreas(r) = 0.05
        Next
    Next
    ANO_REFERENCIA = year(Now)
    ReDim NomeRegiao(NREGIONS)
    NomeRegiao(1) = "NORTE LITORAL"
    NomeRegiao(2) = "INTERIOR"
    NomeRegiao(3) = "CENTRO NORTE"
    NomeRegiao(4) = "CENTRO SUL"
    NomeRegiao(5) = "SUL"
    Call read_rankings
    Call read_conc_codes
End Sub
```

```
        Public Sub read_database()
            Dim s As String
            Dim id As String, conc As Long, regiao As Long, estrato As Long
            Dim idade  As Single, area As Single, rot As Long, n_pequenas As Single
            Dim nm5 As Single, hd As Single, ab As Single, dqm As Single, iqe As Single
            Dim vol_merc As Single, i As Long, aux As Boolean
            NStands = 0
            'primeiro contar os povs
            Open DATABASE_NAME For Input As #1
            Line Input #1, s
            While Not EOF(1)
                Input #1, id, conc, regiao, estrato, idade, area, rot, n_pequenas, nm5, hd,
ab, dqm, iqe, vol_merc
                If n_pequenas + nm5 >= 0 Then NStands = NStands + 1
            Wend
            Close #1
            'agora podemos de facto carregar os povoamentos
            Open DATABASE_NAME For Input As #1
            ReDim Stands(NStands)
            i = 1
            Line Input #1, s
            While Not EOF(1)
                Input #1, id, conc, regiao, estrato, idade, area, rot, n_pequenas, nm5, hd,
ab, dqm, iqe, vol_merc
                'atenção!!!!!!!!!!
                If iqe = 0 Then iqe = Regions(regiao).iqe
                If id = "399" Then
                    id = id
                End If
                If n_pequenas + nm5 >= 0 Then
                    Stands(i).age = idade
                    Stands(i).age_init = idade
                    If idade = 0 Then
                        Stands(i).age_known = True
                    Else
                        Stands(i).age_known = False
                    End If
                    Stands(i).area = area
                    Stands(i).ba = ab
                    Stands(i).ba_init = ab
                    Stands(i).dqm = dqm
                    Stands(i).hd = hd
                    Stands(i).hd_init = hd
                    Stands(i).id = id
                    Stands(i).iqe = iqe
                    Stands(i).narvs = nm5 + n_pequenas
                    Stands(i).narvs_init = nm5 + n_pequenas
                    Stands(i).narvs_plant = Regions(regiao).narvs_plant
                    Stands(i).nm5 = nm5
                    Stands(i).nm5_init = nm5
                    Stands(i).region = regiao
                    Stands(i).concelho = conc
                    Stands(i).rot = rot
                    Stands(i).vol_sc = vol_merc

                    'initialization routines...
                    'these should NEVER run!!!!!!!
                    If hd = 0 Then
                        aux = Stands(i).age_known
                        Stands(i).age_known = True
                        Stands(i).hd  =  calc_hdom(Stands(i),  ab,  hd)  'call  the  globulus
function!!!!
                        Stands(i).age_known = aux
                    End If
                    If ab = 0 Then
                        aux = Stands(i).age_known
                        Stands(i).age_known = True
                        Stands(i).ba  =  calc_abas(Stands(i),  ab,  hd,  nm5)  'call  the  globulus
function!!!!
                        Stands(i).age_known = aux
                    End If
                    If n_pequenas + nm5 = 0 Then
                        aux = Stands(i).age_known
                        Stands(i).age_known = True
```

```
                Stands(i).nm5 = calc_narvs_n(Stands(i))
                Stands(i).nm5_init = Stands(i).nm5
                Stands(i).narvs = Stands(i).nm5
                Stands(i).narvs_init = Stands(i).nm5
                Stands(i).age_known = aux
                '!!!!!!!!!!!! - Rot is used as a flag to avoid processing of
                ' strange areas (n_arvs=0
                If Stands(i).age > 0 Then
                    Stands(i).rot = 0
                    Stands(i).ba = 0
                    Stands(i).hd = 0
                    Stands(i).vol_sc = 0
                    Stands(i).vol_cc = 0
                End If
            End If

            i = i + 1
        End If
    Wend
    Close #1
End Sub

Public Sub read_iqes(fname As String)
    Dim s As String
    Dim reg As Long, iqe As Single
    Open fname For Input As #1
    Line Input #1, s 'get the header
    While Not EOF(1)
        Input #1, reg, iqe
        Regions(reg).iqe = iqe
    Wend
    Close #1
End Sub

Public Sub read_narvs_plant(fname As String)
    Dim s As String
    Dim reg As Long, n_plant As Single
    Open fname For Input As #1
    Line Input #1, s 'get the header
    While Not EOF(1)
        Input #1, reg, n_plant
        Regions(reg).narvs_plant = n_plant
    Wend
    Close #1

End Sub

Public Sub read_concelhos(fname As String)
    'lê as proporções de áreas por concelho em cada região
    Dim s As String
    Dim conc As Long, reg As Long, prop As Single, max_conc As Long, c As Long
    Open App.Path & "\" & fname For Input As #1
    Line Input #1, s 'get the header
    While Not EOF(1)
        Input #1, conc, reg, prop
        Regions(reg).NCONCELHOS = Regions(reg).NCONCELHOS + 1
        ReDim Preserve Regions(reg).Concelhos(Regions(reg).NCONCELHOS)
        ReDim Preserve Regions(reg).prop_area(Regions(reg).NCONCELHOS)
        Regions(reg).Concelhos(Regions(reg).NCONCELHOS) = conc
        Regions(reg).prop_area(Regions(reg).NCONCELHOS) = prop
        If conc > max_conc Then max_conc = conc
    Wend
    Close #1
    'actualizar a tabela dos concelhos
    NCONCELHOS = max_conc
    ReDim BurntConcelhos(NCONCELHOS)
    Call actualiza_results
    For reg = 1 To NREGIONS
        For c = 1 To Regions(reg).NCONCELHOS
            conc = Regions(reg).Concelhos(c)
            BurntConcelhos(conc).regiao = reg
        Next
    Next
End Sub
```

```
Public Sub read_conc_codes()
    'lê os códigos de concelho! atenção que a BD deve ligar aos CODs
    'e não aos IDs, com está neste momento!!!!!!!
    'o ficheiro para jä tem de estar ordenado por ordem crescente de IDs'!!!!!
    Dim s As String
    Dim id As Long, cod As Long, nome As String, reg_glob As Long
    Open App.Path & "\concelhos.csv" For Input As #1
    Line Input #1, s
    ReDim ConcCodes(1)
    While Not EOF(1)
        Input #1, id, nome, cod, reg_glob
        ReDim Preserve ConcCodes(id)
        ConcCodes(id).nome = nome
        ConcCodes(id).cod = cod
        ConcCodes(id).reg_glob = reg_glob
    Wend
    NConcCodes = id
    Close #1


End Sub

Public Sub read_rankings()
    Dim reg As Long, r As Long, rank As Single
    Dim idade As Long, dqm As Single, s As String
    Open "r_regiao.csv" For Input As #1
    Line Input #1, s
    For r = 1 To NREGIONS
        Input #1, reg, rank
        RankRegiao(reg) = rank
    Next
    Close #1

    Open "r_idade.csv" For Input As #1
    Line Input #1, s
    While Not EOF(1)
        Input #1, idade, rank
        RankIdade.nclasses = RankIdade.nclasses + 1
        ReDim Preserve RankIdade.maxclass(RankIdade.nclasses)
        ReDim Preserve RankIdade.valor(RankIdade.nclasses)
        RankIdade.maxclass(RankIdade.nclasses) = idade
        RankIdade.valor(RankIdade.nclasses) = rank
    Wend
    Close #1

    Open "r_dqm.csv" For Input As #1
    Line Input #1, s
    While Not EOF(1)
        Input #1, dqm, rank
        RankDQM.nclasses = RankDQM.nclasses + 1
        ReDim Preserve RankDQM.maxclass(RankDQM.nclasses)
        ReDim Preserve RankDQM.valor(RankDQM.nclasses)
        RankDQM.maxclass(RankDQM.nclasses) = dqm
        RankDQM.valor(RankDQM.nclasses) = rank
    Wend
    Close #1

End Sub

Public Sub actualiza_results()
    Dim y As Long
    For y = 1 To NYEARS
        ReDim Resultados(y).Concelhos(NCONCELHOS)
    Next
End Sub

Public Sub write_results()
    Dim y As Long, c As Long, s As Long
    Dim res As Result, st As Stand
    Open SIM_PATH & "\geral.csv" For Output As #1
    Open SIM_PATH & "\concelhos.csv" For Output As #2
    Open SIM_PATH & "\areas_cort.csv" For Output As #3
    Open SIM_PATH & "\fogos.csv" For Output As #4
```

```
            Open SIM_PATH & "\areas_novas.csv" For Output As #5
            Print #1, "Ano, VolCort, Stock, ArTot, ArReconv, ArAband, ArArd, VolArd, ArNova,
ArCort, NCorts, IdMed, IdMedCort"
            Print #2, "Ano, Concelho, ArCort, ArAband, ArReconv, NCorts, Stock, VolCort,
IdMed, IdMedCort"
            Print #3, "Ano, ID, Regiao, Concelho, Area, IQE, Idade, VolCort"
            Print #4, "Ano, Concelho, ArArdida, VolArdido"
            Print #5, "Ano, Concelho, ArNova"
            For y = 1 To NYEARS
                Write   #1,   y   +   ANO_REFERENCIA,   HStands(y).vol_sc,   HStands(y).stock,
HStands(y).area_em_pe, HStands(y).area_reconv, HStands(y).area_aband, _
                            HStands(y).area_ardida,                 HStands(y).volume_ardido,
HStands(y).area_nova, HStands(y).area_cort, HStands(y).st_harv, _
                            HStands(y).idade_med, HStands(y).idade_med_corte
                For c = 1 To NCONCELHOS
                    res = Resultados(y).Concelhos(c)
                    If Resultados(y).Concelhos(c).st_harv > 0 Then
                        Write #2, y + ANO_REFERENCIA - 1, ConcCodes(c).nome, res.area_cort,
res.area_aband, res.area_reconv, _
                             res.st_harv,     res.stock,     res.vol_cort,     res.idade_med,
res.idade_med_cort
                    End If
                    If Resultados(y).Concelhos(c).area_ardida > 0 Then
                        Write #4, y + ANO_REFERENCIA - 1, ConcCodes(c).nome, res.area_ardida,
res.vol_ardido
                    End If
                    If res.area_nova > 0 Then
                        Write #5, y + ANO_REFERENCIA - 1, ConcCodes(c).nome, res.area_nova
                    End If
                Next
                For s = 1 To HStands(y).st_harv
                    st = HStands(y).Stands(s)
                    Write    #3,    y    +    ANO_REFERENCIA    -    1,    st.id,    st.region,
ConcCodes(st.concelho).nome, st.area, st.iqe, st.age, st.vol_sc * st.area
                Next
            Next
            Close #1, #2, #3, #4, #5

        End Sub
```